

PAT-NO: JP404052844A

DOCUMENT-IDENTIFIER: JP 04052844 A

TITLE: COMMUNICATION PROTOCOL PROCESSING SYSTEM

PUBN-DATE: February 20, 1992

INVENTOR-INFORMATION:

NAME

TSUCHIYA, YUKIHISA

KISHIMOTO, GIICHI

ASSIGNEE-INFORMATION:

NAME

COUNTRY

HITACHI LTD

N/A

NIPPON TELEGR & TELEPH CORP <NTT> N/A

APPL-NO: JP02155220

APPL-DATE: June 15, 1990

INT-CL (IPC): G06F013/00, H04L029/06

ABSTRACT:

PURPOSE: To minimize the change of communication protocol processing at the time of changing an application layer protocol by preparing one conversion information table and two processing parts and executing encoding/decoding processing.

CONSTITUTION: In the case of executing encoding processing, a conversion control part receives an internal expression table area and a table A from an encoding processing requesting part, determines a processing-sorted transfer syntax A and finds out P record ID = 01 corresponding to a transfer syntax A in

a protocol data conversion information table. In the case of executing decoding processing, the conversion control part receives the transfer syntax, the internal expression table area and the table A from a decoding processing requesting source, points out the head of the transfer syntax and finds out leading P record **ID** = 01 in the protocol data conversion information table.

COPYRIGHT: (C)1992,JPO&Japio

⑫ 公開特許公報(A) 平4-52844

⑮ Int. Cl.³G 06 F 13/00
H 04 L 29/06

識別記号

3 5 3 C

庁内整理番号

7368-5B

⑬ 公開 平成4年(1992)2月20日

8020-4M H 04 L 13/00 3 0 5 B

審査請求 未請求 請求項の数 1 (全7頁)

⑭ 発明の名称 通信プロトコル処理方式

⑯ 特 願 平2-155220

⑰ 出 願 平2(1990)6月15日

⑱ 発 明 者 土 谷 往 久 神奈川県横浜市戸塚区戸塚町5030番地 株式会社日立製作所ソフトウェア工場内

⑲ 発 明 者 岸 本 義 一 東京都千代田区内幸町1丁目1番6号 日本電信電話株式会社内

⑳ 出 願 人 株式会社日立製作所 東京都千代田区神田駿河台4丁目6番地

\(出 願 人 日本電信電話株式会社 東京都千代田区内幸町1丁目1番6号

\(代 理 人 弁理士 小川 勝男 外1名

明 細 書

〔従来の技術〕

1. 発明の名称

通信プロトコル処理方式

2. 特許請求の範囲

1. ISO-OSI 規定の抽象構文定義に従った通信プロトコルにより通信するシステムにおいて、プロトコルデータ(転送構文)の各要素の情報をもつプロトコルデータ変換情報テーブルと、該プロトコルデータ変換情報テーブルに従い転送構文の要素毎に転送構文と抽象構文とを変換する機構を設けることにより、応用層プロトコルの変更に対して通信プロトコル処理の変更を極小化できることを特徴とするプレゼンテーション層の通信プロトコル処理方式。

3. 発明の詳細な説明

〔産業上の利用分野〕

本発明は、国際標準である開放型システム間相互接続(OSI規格)に基づくコンピュータ間通信における、プレゼンテーション層の通信プロトコル処理方式に関するものである。

OSI 参照モデルのプレゼンテーション層の役割は、応用層が扱うデータ表現(コード体系、数値の精度等)の相違を吸収するためにある。応用層が扱うデータ表現を抽象構文、プレゼンテーション層以下が扱うデータ表現を転送構文と呼ぶ。抽象構文と転送構文の関係が、OSI 規格の抽象構文記述方法1(ASN.1)として規定される。ASN.1 規定に基づき、転送構文形式から抽象構文形式への変換処理(デコード処理)、および抽象構文形式から転送構文形式への変換処理(エンコード処理)が、実現される。

従来、応用層処理の複雑さから、応用層処理部が扱うデータ形式は、ASN.1 規定である抽象構文形式そのものではなく、応用層処理部に都合のよい内部表現形式(テーブル形式)としている。従って、プレゼンテーション層の通信プロトコル処理、すなわち転送構文形式と内部表現形式との間のエンコード/デコード処理が、応用層プロトコルの抽象構文定義に基づく、交換手続き処理と

して実現される（第5図）。

デコード処理手続きおよびデコード処理手続きを例により説明する。抽象構文の定義例を第2図、定義に基づく転送構文および内部表現形式のテーブルの構成をそれぞれ第3図および第4図に示す。なお、本文中の T_n , L_n , V_n (n は数字)は第3図中にのものに対応する。

〈エンコード処理手続き〉

①) 内部表現形式のテーブル（テーブルA）の構成要素の値と長さを求める。

①-①) Xの値(V_2)と長さ(L_2)を求める。

①-②) Z1の値(V_4)と長さ(L_4)を求める。

①-③) Z2の値(V_5)と長さ(L_5)を求める。

②) Z2の転送構文を作成する。

②-①) Z2の属性(T_5), 長さ(L_5)および値(V_5)の順に並べた転送構文を作成する。

③) Z1の転送構文の作成

③-①) Z1の属性(T_4), 長さ(L_4)および値(V_4)の順に並べた転送構文を作成する。

④) Yの転送構文のを作成する。

チェックする。

②-②) ポインタを L_1 域の次に移動する。

⑤) XのデータをテーブルAに設定する。

⑤-①) ポインタの指す内容, 属性(T_2), をチェックし, ポインタの次の内容から, V_2 の長さ(L_2)を得る。

⑤-②) ポインタを L_2 域の次に移動する。

⑤-③) ポインタの指す内容を V_2 の長さ分, 値(V_2), を取り出し, テーブルAに設定する。

⑤-④) ポインタを V_2 域の次に移動する。

⑥) Yのデータの属性を確認する。

⑥-①) ポインタの指す内容, 属性(T_3), をチェックし, ポインタの次の内容から, V_3 の長さ(L_3)を得る。

⑥-②) ポインタを L_3 域の次に移動する。

⑦) Z1のデータをテーブルAに設定する。

⑦-①) ポインタの指す内容, 属性(T_4), をチェックし, ポインタの次の内容から, V_4 の長さ(L_4)を得る。

④-①) Z1とZ2の転送構文の長さの和, Yの長さ(L_3), を求める。

④-②) Yの属性(T_3), 長さ(L_3), Z1の転送構文およびZ2の転送構文の順に並べた転送構文を作成する。

⑤) Xの転送構文を作成する。

⑤-①) Xの属性(T_2), 長さ(L_2)および値(V_2)の順に並べた転送構文を作成する。

⑥) Aの転送構文を作成する。

⑥-①) XとYの転送構文の長さの和, Aの長さ(L_1), を求める。

⑥-②) Aの属性(T_1), 長さ(L_1), Xの転送構文およびYの転送構文の順に並べた転送構文を作成する。

⑦) エンコード処理を終了する。

〈デコード処理手続き〉

①) 受信した転送構文域の先頭をポインタとして設定する。

②) Aデータの属性を確認する。

②-①) ポインタの指す内容, 属性(T_1), をチ

⑤-②) ポインタを L_4 域の次に移動する。

⑤-③) ポインタの指す内容を V_4 の長さ分, 値(V_4), を取り出し, テーブルAに設定する。

⑤-④) ポインタを V_4 域の次に移動する。

⑥) Z2のデータをテーブルAに設定する。

⑥-①) ポインタの指す内容が転送構文の最後であるならば, デフォルト値をテーブルAに設定し⑦へ進む。そうでなければ, ⑥-②以降を行う。

⑥-②) ポインタの指す内容, 属性(T_5), をチェックし, ポインタの次の内容から, V_5 の長さ(L_5)を得る。

⑥-③) ポインタを L_5 域の次に移動する。

⑥-④) ポインタの指す内容を V_5 の長さ分, 値(V_5), を取り出し, テーブルAに設定する。

⑦) デコード処理を終了する。

[発明が解決しようとする課題]

転送構文の各要素は, 第3図に示す通りの[属

性、長さ、値]の3つ組構成であり、値は基本要素か下位レベルの3つ組構成のいずれからなる。応用層プロトコルの変更により、あるレベルの3つ組構成が変更されると、それを包含する3つ組の構成がすべて変更となる。従来の手続き処理によるエンコード/デコード処理では、応用層プロトコルの改訂に伴い、あるレベルの3つ組構成が変更されると、それを包含する3つ組構成に対する手続き処理を変更する必要が生じ、変更箇所が広範囲にわたることになる。

〔課題を解決するための手段〕

本発明では、以下の1つの変換情報テーブルと2つの処理部を設け、エンコード/デコード処理を実現する(第1図)。

●ASN.1に基づき定義された応用層プロトコルの転送構文の構造を示すプロトコルデータ変換情報テーブル

これは、転送構文の各要素対応の変換情報をレコードとして持つ。各レコードは以下の情報からなる。

- ・レコード識別子(ID)
- ・同一レベルの次の3つ組のレコード識別子(NP)
- ・下位レベルの次の3つ組のレコード識別子(DP)
- ・属性コード(TP)
- ・OPTIONAL情報(OP)
- ・デフォルト情報(DF)
- ・内部表現テーブル内オフセット情報(OF)
- ・IMPLICIT情報(IM)
- ・タグ情報(TG)

●ASN.1が規定する属性(INTEGER, IASString, SEQUENCE, CHOICE等)に対応したエンコード処理およびデコード処理を行う属性対応処理部

●プロトコルデータ変換情報テーブルに基づき、属性対応処理部を制御し、転送構文と内部表現形式の抽象構文(内部表現テーブル)とを変換する変換制御部

プロトコル変換情報テーブル、属性対応処理部および変換制御部を使用したエンコード/デコード処理の流れを示す。

〈エンコード処理〉

①) 変換制御部が、エンコードする内部表現テーブルをエンコード処理依頼元から受け取り、内部表現テーブルに指定された処理種別(作成する転送構文)を決定する。

②) 変換制御部が処理種別に基づき、プロトコルデータ変換情報テーブルの対応するレコード(Pレコード)を見いだす。

③) 変換制御部が、PレコードのTP情報に対応する属性対応処理部を呼び出す。

④) 属性対応処理部が、属性に基づいた属性対応の転送構文を作成する。

④-①) 属性が基本要素の場合、内部表現テーブルの先頭からPレコードの情報のOF値目が指す領域の情報を値として、転送構文を作成する。

④-②) 属性が基本要素でない場合、DP情報が指すPレコードを求め、変換制御部を再起的に呼び出し、下位レベルの転送構文を作成する(③を呼び出す)。次に、作成された下位レベルの転送構文

を値とした転送構文を作成する。

⑤) 変換制御部が、④で作成された転送構文を作成済みの転送構文の次に追加する。

⑥) 変換制御部が、PレコードのNP情報に従い、次のPレコードを見いだす。次のPレコードがある場合、③に戻る。そうでない場合、変換制御部の呼び出し元に転送構文を返却する。

〈デコード処理〉

①) 変換制御部が、転送構文と内部表現テーブル域をデコード処理依頼元から受け取る。転送構文の先頭をポインタに設定する。

②) 変換制御部がプロトコルデータ変換情報テーブルの先頭のPレコードを見いだす。

③) 変換制御部が、PレコードのTP情報あるいはTG情報とポインタが指す転送構文の属性が一致するかを調べる。

③-①) TP情報あるいはTG情報と一致する場合、変換制御部がPレコードが示す属性に対応する属性対応処理部を明示モードで呼び出す。

- ③-②) 一致せずかつDP情報が真の場合、変換制御部がPレコードの示す属性に対応する属性対応処理部を暗示モードで呼び出す。
- ③-③) 上記以外の場合、PレコードのNP情報に従い、次のPレコードを見だし、再度③を行う。
- ④) 属性対応処理部が、属性に基づき、属性対応の転送構文を作成する。
- ④-①) 明示モードかつ属性が基本要素の場合、転送構文中の値の長さと値を求め、内部表現テーブルの先頭からPレコードの情報のOF値目の領域に設定する。
- ④-②) 明示モードかつ属性が基本要素でない場合、ポインタが指す転送構文の値とPレコードのDP情報が指すPレコードを見だし、変換制御部を再起的に呼び出し(③を呼び出し)、下位レベルの転送構文から内部表現テーブルに情報を設定する。
- ④-③) 暗示モードかつ属性が基本要素の場合、PレコードのDP情報を、内部表現テーブルの先頭からPレコードの情報のOF値目の領域に設定する。
- ④-④) 暗示モードかつ属性が基本要素でない場合、PレコードのDP情報が指すPレコードを見だし、変換制御部を再起的に呼び出し(③を呼び出し)、下位レベルの転送構文から内部表現テーブルに情報を設定する。
- ⑤) 変換制御部が、PレコードのNP情報より次のPレコードを見だす。次のPレコードがある場合、④に戻る。そうでない場合、変換制御部の呼び出し元に戻る。
- 〔作用〕
- 上述の構成および処理手順により、応用層プロトコルの変更に対する対応がプロトコルデータ変換情報テーブルの変更のみで済み、変更箇所が極小化できる。
- 〔実施例〕
- (IA5String処理)を呼び出す。
- ④-②) 属性対応処理部(IA5String処理)は、属性が基本要素であるため、Aテーブルの先頭からOF値目の値(V2)とその長さ(L2)を取り出し、データXの転送構文を作成し、変換制御部に返却する。
- ④-③) 変換制御部が、作成されたXの転送構文を保持する。
- ④-④) 変換制御部が、NP情報が指すPレコード(ID=03)を見だす。
- ④-⑤) 変換制御部が、Pレコード(ID=03)のTP情報(30)に対応する属性対応処理部(SEQUENCE処理)を呼び出す。
- ④-⑥) 属性対応処理部(SEQUENCE処理)は、属性が基本要素でないため、DP情報が指すPレコード(ID=04)を求め、再起的に変換制御部を呼び出す。
- ④-⑥-①) 変換制御部が、Pレコード(ID=04)のTP情報(02)に対応する属性対応処理部(INTEGER処理)を呼び出す。
- エンコード処理およびデコード処理の実施例を、従来の技術で示した例により示す。本文中の T_n , L_n , V_n (n は数字)は第3図に示されるものに対応する。
- 〈エンコード処理の実施例〉
- ①) 変換制御部が、内部表現テーブル域、テーブルA(第4図)、をエンコード処理依頼部から受取り、処理種別(転送構文A)を決定する。
- ②) 変換制御部が、プロトコルデータ変換情報テーブルの転送構文Aに対応するPレコード(ID=01)を見だす。
- ③) 変換制御部が、Pレコード(ID=01)のTP情報(30)に対応する属性対応処理部(SEQUENCE処理)を呼び出す。
- ④) 属性対応処理部(SEQUENCE処理)は、属性が基本要素でないため、DP情報が指すPレコード(ID=02)を求め、変換制御部を再起的に呼び出す。
- ④-①) 変換制御部が、Pレコード(ID=02)のTP情報(16)に対応する属性対応処理部

④-⑥-②) 属性対応処理部(INTEGER処理)は、属性が基本要素であるため、Aテーブルの先頭から0F値目の値(V4)とその長さ(L4)を取り出し、Z1のデータの転送構文を作成し、変換制御部に返却する。

④-⑥-③) 変換制御部が、作成されたZ1の転送構文を保持する。

④-⑥-④) 変換制御部が、NP情報が指すプレコード(ID=05)を見いだす。

④-⑥-⑤) 変換制御部が、プレコード(ID=05)のTP情報(16)に対応する属性対応処理部(IA5String処理)を呼び出す。

④-⑥-⑥) 属性対応処理部(INTEGER処理)は、属性が基本要素であるため、Aテーブルの先頭から0F値目の値(V5)とその長さ(L5)を取り出し、Z2のデータの転送構文を作成し、変換制御部に返却する。

④-⑥-⑦) 変換制御部が、保持されたZ1に

作成されたZ2の転送構文を追加し保持する。

④-⑥-⑧) 変換制御部が、NP情報が指すプレコードがないため、呼び出し元に転送構文を返却する。

④-⑦) 変換処理部から返却された転送構文を値(V3)とし、Yのデータの転送構文を作成し、変換制御部に返却する。

④-⑧) 変換制御部が、保持されたXに作成されたYの転送構文を追加し保持する。

④-⑨) 変換制御部が、プレコード(ID=03)のNP情報が指すプレコードがないため、呼び出し元に転送構文を返却する。

⑤) 変換処理部から返却された転送構文を値(V1)とし、Aのデータの転送構文を作成し、変換制御部に返却する。

⑥) 変換制御部が、作成されたAの転送構文を保持する。

⑦) 変換制御部が、プレコード(ID=01)のNP情報が指すプレコードがないため、呼び出し元に転

送構文を返却する。

〈デコード処理の実施例〉

①) 変換制御部が、転送構文(第3図)と内部表現テーブル域、テーブルA(第4図)、をデコード処理依頼元から受け取る。転送構文の先頭をポイントする。

②) 変換制御部が、プロトコルデータ変換情報テーブルの先頭プレコード(ID=01)を見いだす。

③) 変換制御部が、プレコード(ID=01)のTP情報(30)とポインタの指す転送構文の属性(30)を比較する。両者が一致するため、変換制御部がプレコード(ID=01)のTP情報(30)に対応する属性対応処理部(SEQUENCE処理)を明示モードで呼び出す。

④) 明示モードかつ非基本要素で呼び出された属性対応処理部が、ポインタから転送構文の値(V1)と、プレコード(ID=01)のDP情報(02)を見いだす。その結果を元に、変換制御部を再起的に呼び出す。

④-①) 変換制御部が、プレコード(ID=02)のTP

情報(16)とポインタの指す転送構文の属性(16)を比較する。両者が一致するため、変換制御部がプレコード(ID=02)のTP情報(16)に対応する属性対応処理部(IA5String処理)を明示モードで呼び出す。

④-②) 明示モードかつ基本要素で呼び出された属性対応処理部が、ポインタの指す転送構文の値(V2)とその長さ(L2)を求め、内部表現テーブルの先頭からプレコード(ID=02)の0F情報(04₁₆)目の領域に設定する。

④-③) 変換制御部が、プレコード(ID=02)のNP情報(03)より、次のプレコード(ID=03)を見いだす。

④-④) 変換制御部が、プレコード(ID=03)のTP情報(30)とポインタの指す転送構文の属性(30)を比較する。両者が一致するため、変換制御部がプレコード(ID=03)のTP情報(30)に対応する属性対応処理

部(SEQUENCE処理)を明示モードで呼び出す。

④-⑤) 明示モードかつ非基本要素で呼び出された属性対応処理部が、ポインタから転送構文の値(V3)と、Pレコード(ID=03)のDP情報(04)を見いだす。その結果を元に、変換制御部を再起的に呼び出す。

④-⑤-①) 変換制御部が、Pレコード(ID=04)のTP情報(02)とポインタの指す転送構文の属性(02)を比較する。両者が一致するため、変換制御部がPレコード(ID=02)のTP情報(02)に対応する属性対応処理部(INTEGER処理)を明示モードで呼び出す。

④-⑤-②) 明示モードかつ基本要素で呼び出された属性対応処理部が、ポインタの指す転送構文の値(V4)とその長さ(L4)を求め、内部表現テーブルの先頭からPレコード(ID=04)の

OF情報(0C...)目の領域に設定する。

④-⑤-③) 変換制御部が、Pレコード(ID=04)のNP情報(05)より、次のPレコード(ID=05)を見いだす。

④-⑤-④) 変換制御部が、Pレコード(ID=05)のTP情報(16)とポインタの指す転送構文の属性(16)を比較する。両者が一致するため、変換制御部がPレコード(ID=05)のTP情報(16)に対応する属性対応処理部(IA5String処理)を明示モードで呼び出す。

④-⑤-⑤) 明示モードかつ基本要素で呼び出された属性対応処理部が、ポインタの指す転送構文の値(V5)とその長さ(L5)を求め、内部表現テーブルの先頭からPレコード(ID=05)のOF情報(10...)目の領域に設定する。

④-⑤-⑥) 変換制御部が、Pレコード(ID=05)の

のNP情報(-)より、次のPレコードがないため、呼び出しに戻る。

④-⑥) 変換制御部が、Pレコード(ID=03)のNP情報(-)より、次のPレコードがないため、呼び出しに戻る。

⑤) 変換制御部が、Pレコード(ID=01)のNP情報(-)より、次のPレコードがないため、呼び出し元(デコード処理依頼元)に戻る。

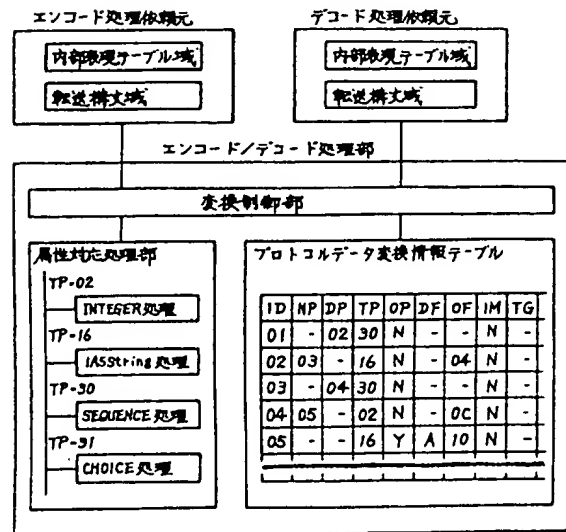
(発明の効果)

以上述べたように、本発明によれば通信プロトコル処理の変更を極小化できる。

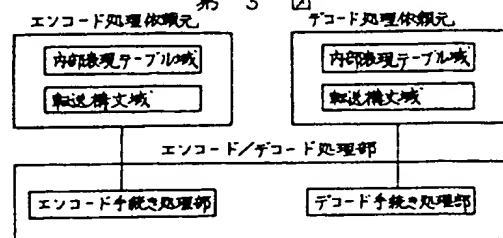
4. 図面の簡単な説明

第1図は本方式の実現例を示す図、第2図は抽象構文の定義例を示す図、第3図は転送構文の例を示す図、第4図は内部表現テーブルの例を示す図、第5図は従来の実現例を示す図である。

第1図



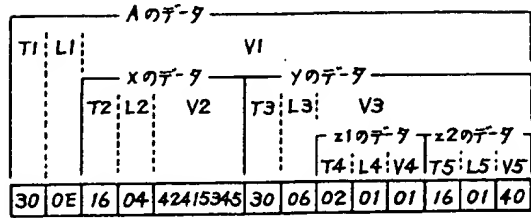
第5図



第 2 図

```
A ::= SEQUENCE
  { x IASstring ,
    y SEQUENCE
      { z1 INTEGER,
        z2 IASstring DEFAULT ('A') OPTIONAL
      }
  }
```

第 3 図



第 4 図

テーブルAの構成

テーブルID	
Xの領域	
Yの領域	Z1の領域
	Z2の領域